Project Overview: Scenario and Review Criteria

Project Overview

You will have the chance to practice and put your skills, knowledge, and abilities to use Node.JS & Express.JS by developing Online Book review application in your final project.

Final Project Scenario

In this project you will assume the role of a back-end developer working for an online retailer selling books. You have been tasked with developing a server-side application that stores, retrieves and manages book ratings and reviews.

Your server-side application is required to provide the following features and capabilities to allow users to:

- Retrieve a list of all books available in the bookshop
- Search for specific books and retrieve their details based on the book's ISBN code, author names and titles
- Retrieve reviews/comments for specified books
- Register as a new user of the application
- Login to the application
- Add a new review for a book (logged in users only)
- Modify a book review (logged in users can modify only their own reviews)
- Delete a book review (logged in users can delete only their own reviews)
- (Multiple users) Access the application at the same time to view and manage different book reviews simultaneously

As is the case with most software development projects, different people in the team work on different parts of the application. Another front-end developer in your team is working on the web-based client-side application that will communicate with your server-side application using REST. Therefore your job is to implement your server-side application as a RESTful web service. A software architect on your team has written the skeleton code for your server-side application using Node.js and Express.js.

To complete the project you will fork the skeleton code into your own repo, clone it locally into your development environment, and develop the code further to implement the CRUD capabilities listed above as HTTP methods in your Express server and test them using cURL/Postman. You will also implement Session and JWT authentication to allow only logged in users to perform certain operations. For your reference, this application comes preloaded with all of the book information.

Furthermore you will need to enhance your code using Promises, Callbacks or Async/Await functions to allow multiple users to interact with the application simultaneously and not have to wait for each other's operations to complete.

Evaluation Criteria – 30 marks total

You can submit your project deliverables in one of the following ways:

- **Option 1: AI-Graded Submission and Evaluation**
  When you choose Option 1, you will be redirected to an AI tool where you can upload your deliverables, which may include URLs, terminal outputs, code snippets, or screenshots. You will then receive an AI-generated grade that will automatically reflect on your progress page.

- **Option 2: Peer-Graded Submission and Evaluation**
  When you choose Option 2, you will upload your deliverables—such as URLs, terminal outputs, code snippets, or screenshots—through the My Submission section. Your submission will then be reviewed either by your peers or by AI grader.

  We recommend using Option 1 for faster grading. However,if you face any issues or cannot access it, you may use Option 2 instead.

  If you encounter any grading problems, please reach out to the Course Team through the Discussion Forums.

Please find the details of the Grading Criteria below:

▼ Criteria for Option 1: AI-Graded Submission and Evaluation:
Please save the specified text (cURL command and its output) for each task in your final assignment, as mentioned in the lab instructions. This project carries 30 points towards your final grade and is weighted as follows:

Note: The repository's JSON file containing the book details will be used for evaluation.

General users:

Task 1: Copy and paste the cURL command and its output, saved as getallbooks, which displays all book(s) retrieved.- **2 Points**

Task 2: Copy and paste the cURL command and its output, saved as getbooksbyISBN, which displays all book(s) retrieved based on the specified ISBN.- **2 Points**

Task 3: Copy and paste the cURL command and its output, saved as getbooksbyauthor, which displays all books retrieved based on the specified author.- **2 Points**

Task 4: Copy and paste the cURL command and its output, saved as getbooksbytitle, which displays all books retrieved based on the specified title.- **2 Points**

Task 5: Copy and paste the cURL command and its output, saved as getbookreview, which displays the initial book review. - **2 Points**

Task 6: Copy and paste the cURL command and its output, saved as register, which displays a message confirming the successful registration of a new user. – **3 Points**

Task 7: Copy and paste the cURL command and its output, saved as login, which displays the result of logging in as a registered user.- **3 Points**

Registered Users:

Task 8: Copy and paste the cURL command and its output, saved as reviewadded, which displays a message and reviews after adding or modifying a book review.- **2 Points**

Task 9: Copy and paste the cURL command and its output, saved as deletereview, which displays a delete message after deleting a book review.- **2 Points**

Node.JS program with 4 methods:

Use Async/Await or Promises with Axios in Node.js for all the four methods.

Task 10: Submit the GitHub URL of the general.js file, which contains the code implementation to retrieve all books and their details based on author, title, and ISBN, using promise callbacks or async/await with Axios.- **8 Points**

Task 11: Copy and paste the cURL command and its output, saved as githubrepo, which shows that your GitHub repository is forked from ibm-developer-skills-network/expressBookReview.- **2 Points**

▶ Criteria for Option 2 : Peer-Graded Submission and Evaluation:

Next Steps

Be sure to read the Overview before starting the step-by-step instructions.

Author(s)

Lavanya T S

Sapthashree K S

Sameep